2001

Application of shunting inhibitory artificial neural networks to medical diagnosis

Ganesh Arulampalam
Edith Cowan University

Abdesselam Bouzerdoum
Edith Cowan University

10.1109/ANZIIS.2001.974056

Application of Shunting Inhibitory Artificial Neural Networks to Medical Diagnosis

G Arulampalam and A Bouzerdoum
Edith Cowan University
100 Joondalup Drive
Joondalup, WA 6027
Australia
g.arulampalam@ecu.edu.au | a.bouzerdoum@ecu.edu.au

Shunting Inhibitory Artificial Neural Networks (SIANNs) are biologically inspired networks in which the neurons interact among each other via a nonlinear mechanism called *shunting inhibition*. Since they are high-order networks, SIANNs are capable of producing complex, nonlinear decision boundaries. In this article, feedforward SIANNs are applied to several medical diagnosis problems and the results are compared with those obtained using *multilayer perceptrons* (MLPs). First, the structure of feedforward SIANNs is presented. Then, these networks are applied to some standard medical classification problems, namely the Pima Indians diabetes and Wisconsin breast cancer classification problems. The SIANN performance compares favourably with that of MLPs. Moreover, some problems with the diabetes dataset are addressed and a reduction in the number of inputs is investigated.

1 Introduction

Shunting Inhibitory Artificial Neural Networks (SIANNs) are biologically inspired networks in which the synaptic interactions among neurons are mediated via a nonlinear mechanism called shunting inhibition, which equips neurons with a gain control mechanism that allows them to operate as adaptive nonlinear filters [1].

Shunting networks have been extensively applied in psychophysics, speech, perception, robotics, adaptive pattern recognition, vision and image processing. Shunting lateral inhibition plays an important role in vision [2, 3]. Cellular neural networks based on shunting inhibition have shown great promise as information processors in vision and image processing tasks [4], but they have not yet been used for classification and function approximation tasks. One of the main reasons for this is the lack of proper training algorithms. Thus far, applications have been limited since the designer has to choose the connection weights based on the task at hand. Another reason is that the operation of shunting inhibitory cellular neural networks is governed by a system of nonlinear differential equations, which must be solved in order to obtain the output of the network for a given input pattern.

The alternative shown here is to operate the shunting networks in a feedforward mode and use the steady-state solution, thereby avoiding the need to obtain a numerical solution for a set of differential equations [1]. This allows the network to operate in a static mode like *multilayer perceptrons* (MLPs). The idea is to exploit the inherent nonlinearity of shunting inhibition to develop powerful, trainable networks, with nonlinear decision surfaces, for classification, nonlinear regression and pattern association.

In this paper, we consider applying SIANNs to medical diagnosis, a form of classification. The process essentially consists of making a diagnosis (classification) from some given symptoms or measurements (inputs) based on past experience. This is the same basic process as classification by a neural network trained using supervised learning. The paper analyses SIANNs and their performance when applied to some benchmark medical diagnosis problems, namely the Pima Indians diabetes and Wisconsin breast cancer datasets. Their performance is also compared to that of MLPs.

The next section describes the structure of the feedforward SIANN and the neuron model used. The third section presents the experiments performed using the SIANN classifier and the results analysed in section 4. Finally, a discussion and conclusion are presented in sections 5 and 6.

2 Shunting Inhibitory Artificial Neural Network Classifier

2.1 Feedforward Network Structure

In shunting inhibitory cellular neural networks (SICNNs), neighbouring neurons exert mutual inhibitory interactions of the shunting type. The activity of each neuron is described by a nonlinear differential equation:

$$\frac{dx_j}{dt} = I_j - a_j x_j - f(\sum_i c_{ji} x_i) x_j + b_j \qquad (1)$$

where x_j represents the activity of the jth neuron; I_j is the external input to the jth neuron; a_j is the passive decay rate of the neuron (positive constant); c_{ji} is the connection weight from the ith neuron to the jth neuron; b_j represents the bias; and f is a non-decreasing activation function.

Equation (1) describes the activity of individual shunting neurons that are arranged in a cellular fashion, where each neuron receives a single external excitatory input and the weighted outputs of the neurons in a predefined neighbourhood are fed back, through the nonlinear activation function, as inhibitory inputs. This cellular form is a recurrent (or feedback) network. The stability of these networks has been investigated by Bouzerdoum and Pinter [2, 5], and they have been successfully applied to vision and image processing tasks [4, 6].

The network being considered in this paper, on the other hand, is a feedforward shunting inhibitory network, where the feedback loops are removed and the inhibitory inputs are now the weighted *external inputs* (I_i) of all the neurons. The shunting neurons are now arranged in a layer (or layers) instead of in a grid. The output layer consists of a layer of perceptrons. This network structure is called a *shunting inhibitory artificial neural network* (SIANN).

Figure 1 shows a feedforward SIANN with a single layer of m shunting inhibitory neurons. The outputs of the shunting neurons are connected to n output neurons (perceptrons).

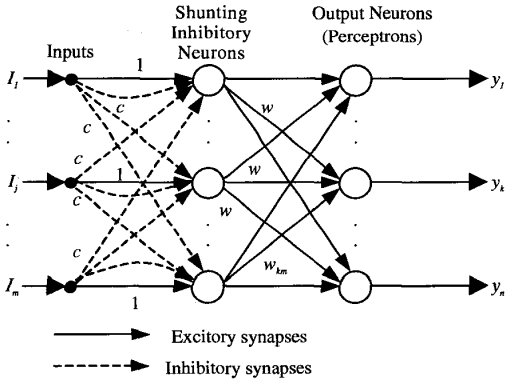

Figure 1: Feedforward SIANN Structure

2.2 Shunting Neuron Model

The differential equation that describes the jth shunting neuron in feedforward network, as shown in Figure 1, is given by:

$$\frac{dx_j}{dt} = I_j - a_j x_j - f(\sum_i c_{ji} I_i) x_j + b_j \qquad (2)$$

In order to study the behaviour of this network in a static mode, the behaviour of the neuron is modelled by its steady-state solution. Using the steady-state solution rather than attempting to solve a set of differential equations simplifies the analysis of the network.

The steady-state solution of equation (2) is given by

$$x_j = \frac{I_j + b_j}{a_j + f\left(\sum_i c_{ji} I_i\right)} \qquad (3)$$

This is shown diagrammatically in Figure 2.

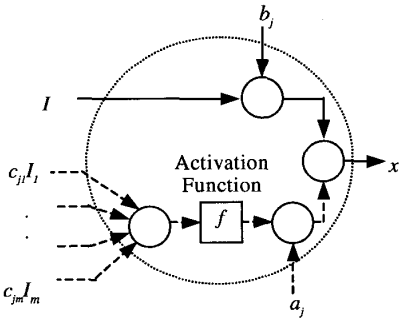

Figure 2: Static Model of a Shunting Neuron

We define the denominator in equation (3) as the shunting term for the jth neuron, s_j, given by

$$s_j \equiv a_j + f\left(\sum_i c_{ji} I_i\right) \qquad (4)$$

The original definition given by equation (1) places constraints that the term a_j be a positive constant and that the activation function f be a positive non-decreasing function. These constraints are now relaxed, and replaced by the constraint that s_j has to be positive so as not to encounter a divide by zero error ie. $s_j > 0$.

The output of the kth output neuron is given by

$$y_k = g(\sum_{j=1}^{m} w_{kj} x_j + b_{ok}) = g(v_k) \qquad (5)$$

where g is the output layer activation function; w_{kj} represents the connection weight from the jth shunting neuron to the kth output neuron; b_{ok} is the bias of output neuron k; and v_k is the input to the kth output neuron.

Due to the nonlinearity introduced by shunting inhibition, a single shunting neuron is capable of producing relatively complex decision boundaries. Figure 3 shows an example of a decision boundary of a single shunting neuron that correctly classifies the XOR problem.

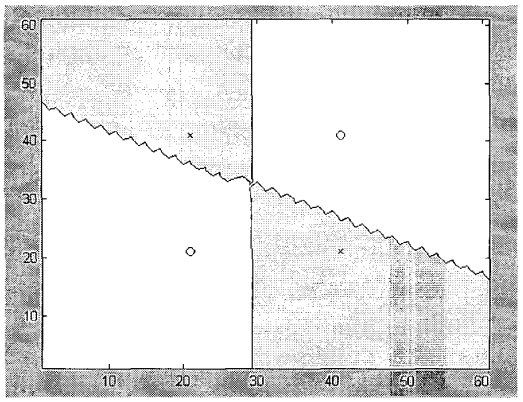

Figure 3: Decision boundary of single shunting inhibitory neuron solving XOR problem

3 Experimental Details

3.1 Classification Problems

Two medical diagnosis problems were used as benchmarks tests for this experiment: the Pima Indians diabetes problem and the Wisconsin breast cancer problem [7]. Both datasets were obtained from the UCI Machine Learning Repository [8]. The diabetes dataset has 768 samples with 8 real-valued inputs and two output classes. The breast cancer dataset has 699 samples with 9 integer inputs and two output classes (benign and malignant). The breast cancer data has missing values that were replaced by zeros before scaling. The diabetes dataset is said not to have any missing values, but there are a number of zero entries that appear to be simply inserted to replace missing values. The impact of these zero values will be discussed later in this paper.

Both dataset inputs have been scaled to the range -1 to 1 for the experiment. Each dataset was partitioned into three sets: the first 50% was used as the training set, the next 25% as the validation set and the last 25% as the test set.

A trained neural network should be able to correctly classify previously unseen inputs. This generalisation ability is the true test of the effectiveness of a classifier or diagnostic tool. The validation set is used for early stopping so that the networks are not overtrained and are able to generalise well. During training the network weights that result in the minimum validation set error are saved. If the validation set error is not reduced for 50 consecutive epochs, the training is stopped and the final network weights used for testing are those that resulted in the minimum validation set error. All results shown are based on the test set, which contains examples not used in the training process.

3.2 Network Structures

The datasets were used to train a number of different SIANN and MLP networks. The number of layers and number of neurons per layer were varied in order to find a pseudo-optimal structure for these problems. The inputs to a SIANN are fed directly into a shunting neuron (refer Figure 1). The basic SIANN structure therefore will have as many shunting neurons as there are inputs. Some structures tested have additional shunting neurons that are fed in with constant zero inputs (dubbed as 'interneurons'). These extra neurons appear to help in classification of complex problems. For the diabetes problem, multilayer SIANNs were also used. Each layer of shunting neurons has the same number of neurons as the previous. In both cases, the output neurons are a single perceptron since both problems have only two classes. For the diabetes problem, two-output SIANNs were also tried where the outputs work on a winner-take-all basis i.e. the output neuron with the largest output is the 'winning' class. The shunting layer neurons had hyperbolic tangent (tansig) activation functions for all cases, while the output layer neurons had either logarithmic (logsig) or linear activation functions.

Two SIANN structures were used for the breast cancer problem: 9-1 (9 shunting neurons, single perceptron output) and 10-1. For the diabetes problem, a number of structures were tried out: 8-1, 9-1, 10-1, 8-2, 10-2, 8-8-1 and 10-10-1.

For the MLPs, a number of different structures were tried with varying number of hidden layer neurons and single output neuron. Only single hidden layer networks were used for the breast cancer problem: 9-1 (9 hidden layer neurons and single output), 6-1 and 12-1. For the diabetes problem both one and two hidden layer MLPs were used: 8-1, 12-1, 6-1, 8-8-1, and 4-2-1.

For each network structure, ten networks with the same structure but with randomly initialised weights were generated. Each set of ten networks was trained using various training algorithms and the results averaged over the ten networks.

3.3 Training Algorithms

The training algorithms developed are based on the standard backpropagation algorithm. The derivation of gradient-based algorithms for SIANNs and their application to simple classification problems such as the parity problem has been shown in earlier work [9].

For this experiment, four training algorithms were used. Two 'standard' algorithms are gradient descent with momentum and adaptive learning rate (GDA) [10] and the Levenberg-Marquadt (LM) algorithm [11]. The third algorithm is a hybrid algorithm (DS-GDA) that updates the shunting neuron weights using the GDA algorithm whereas the output (perceptron) layer weights are updated by direct solution using a linear least squares method [12].

The fourth algorithm is based on a recurrent neural network which was first proposed by Bouzerdoum and Pattison [13] for bound constrained quadratic optimisation. In this Quadratic Neural Network (QNN) algorithm, a recursive equation is used to represent the 'recurrent neural network' that gives the optimal weight update. The derivation of this algorithm, and its application to multilayer perceptrons, are given elsewhere [14].

When training SIANNS, the parameter a_j has to be constrained so that the shunting term s_j, as given in equation (4), always remains positive. The QNN algorithm was developed because it has the ability to incorporate such constraints while working out the optimal weight update.

4 Results and Analysis

4.1 Results for Breast Cancer Diagnosis

The breast cancer problem appears to be quite simple for the neural networks. The worst combination of structure and training algorithm had on average test set error of only 3%. Most cases had average error under 1% with a number of the runs achieving 100% test success. The SIANNs were achieving 3 to 6 runs out of 10 with no error whereas the MLPs got a maximum of 3 runs in a set achieving this. Summarised results for the best structures are shown in Table 1.

The results show that the networks can be trained fast. If the extra 50 epochs for validation stop are discounted, the effective training time is generally under 100 epochs. The only exception is the GDA algorithm, where in some runs the validation check doesn't stop training for a long time because the validation error is reduced very slowly.

Overall, SIANNs achieved slightly better results with lower average error and more networks being able to achieve 100% success. The results, however, depend not only on the structure of the network but the algorithm used to train them.

Table 1: Results for Breast Cancer dataset

Training Algorithm	Activation functions		Runs with no error	Average Epochs		Average Test Error	
	Hidden	Output		All runs	No error	% error	(std. dev.)
SIANN 9-1							
LM	Tansig	Logsig	3	66	65	1.02	(0.88)
GDA	Tansig	Logsig	3	3156	224	0.40	(0.43)
DS-GDA	Tansig	Logsig	5	143	167	0.28	(0.30)
DS-GDA	Tansig	Linear	1	69	51	0.71	(0.50)
QNN	Tansig	Logsig	6	82	75	0.23	(0.29)
MLP 9-1							
LM	Tansig	Linear	1	56	62	1.02	(1.64)
GDA	Tansig	Linear	3	226	270	0.40	(0.27)
QNN	Tansig	Linear	3	150	129	0.40	(0.27)

Table 2: Results for Diabetes dataset with all 8 inputs

Training Algorithm	Activation functions		Runs error <20%	Avg. epochs (all runs)	Best case % test error	Average Test Error	
	Hidden	Output				% error	(std. dev.)
SIANN 8-1							
LM	Tansig	Logsig	2	74	19.27	21.09	(1.26)
GDA	Tansig	Logsig	0	1207	20.31	24.84	(6.91)
DS-GDA	Tansig	Logsig	6	61	18.75	19.95	(0.70)
DS-GDA	Tansig	Linear	4	54	18.75	20.10	(0.50)
QNN	Tansig	Logsig	3	120	19.79	21.15	(1.26)
SIANN 8-2							
LM	Tansig	Logsig	5	143	18.23	19.84	(0.87)
GDA	Tansig	Logsig	4	238	19.27	21.09	(1.58)
DS-GDA	Tansig	Logsig	7	136	17.71	20.05	(1.86)
DS-GDA	Tansig	Linear	4	56	18.23	21.56	(3.74)
QNN	Tansig	Logsig	2	136	17.71	20.89	(1.50)
MLP 8-1							
GDA	Tansig	Logsig	3	177	18.75	20.52	(0.93)
QNN	Tansig	Linear	5	171	19.79	20.62	(1.08)
MLP 6-1							
LM	Tansig	Linear	3	56	17.71	20.16	(1.32)

4.2 Results for Diabetes Diagnosis

The diabetes diagnosis problem was a lot harder for the neural networks compared to the breast cancer problem. The average test error was generally between 20% to 22%, with best case results as low as 17.7%. This compares well with previously reported results for this problem [15, 16] where the best case results were above 20%. A number of different structures were tried, but the

results were not very different. The summarised results for the best-performing structures are given in Table 2.

From the results, it can be seen that both SIANNs and MLPs can consistently achieve about 80% success rate for diabetes prediction, from the given inputs. The difference in performance between SIANNs and MLPs is not significant, but the SIANNs are able to achieve slightly lower error rates overall.

4.3 Modifying the Diabetes Diagnosis Problem

The results obtained above indicate a limit to the ability of the neural networks to correctly predict the incidence of diabetes, based on the input data. The question then arose, how good is the input data? In section 2.1, it was mentioned that there are a number of spurious zero values in the data. These were left in as representative of imperfect real world data or 'noise'. However, closer inspection of the data revealed that two parameters had large proportions of nonsensical zero values. The parameter *triceps skin fold thickness (mm)* (TSFT) had 227 out of 768 samples with a zero value, with the proper values ranging from 7 to 99. The parameter *2-hour serum insulin (mu U/ml)*, (2-SI) on the other hand, with proper values ranging from 14 to 846, had 374 out of 768 samples with zero! This meant that these two parameters have approximately 30% and 50% of samples with missing values. It has been noted by Waschulzik et al [17] that this creates spurious connections between otherwise unrelated samples.

The effect of the missing values in these two parameters on the classifier performance was then investigated using two methods. In the first case, the missing values were set to the mean of the non-zero values and then the data was scaled and partitioned as before. In the second case, the two parameters concerned were eliminated totally. The neural networks were trained using only the remaining six inputs. For network structures that had been used earlier, the networks were set to the same initial conditions. These results are presented in Tables 3 and 4 respectively.

The results show changing the missing values from zero to the mean for the two input parameters TSFT and 2-SI does not significantly affect the performance of the classifiers, but neither does removing these two inputs totally. Removing them has the advantage that the same results can now be obtained using smaller networks. In fact, the best results achieved were with an 8-1 SIANN trained with 6 inputs (16.67% error).

Table 3: Results for Diabetes Dataset with Missing Values for TSFS and 2-SI Set to Mean Value

Training Algorithm	Activation functions: Hidden	Activation functions: Output	Runs error <20%	Avg. epochs (all runs)	Best case % test error	Average test error: % error	Average test error: (std. dev.)
SIANN 8-1							
LM	Tansig	Logsig	1	82	19.79	21.72	(0.92)
GDA	Tansig	Logsig	1	1175	19.79	24.69	(7.37)
DS-GDA	Tansig	Logsig	4	217	19.27	20.78	(1.24)
DS-GDA	Tansig	Linear	4	54	18.75	20.57	(1.21)
QNN	Tansig	Logsig	3	127	19.79	21.04	(1.05)

Table 4: Results for Diabetes Dataset with TSFT and 2-SI Inputs Removed (6 Inputs)

Training Algorithm	Activation functions: Hidden	Activation functions: Output	Runs error <20%	Avg. epochs (all runs)	Best case % test error	Average test error: % error	Average test error: (std. dev.)
SIANN 8-1							
LM	Tansig	Logsig	0	72	20.31	21.30	(0.79)
GDA	Tansig	Logsig	0	158	20.31	26.04	(7.85)
DS-GDA	Tansig	Logsig	4	151	18.23	20.31	(1.10)
DS-GDA	Tansig	Linear	3	52	19.79	29.95	(18.6)
QNN	Tansig	Logsig	6	146	16.67	19.95	(1.72)
SIANN 6-1							
LM	Tansig	Logsig	2	80	18.23	20.83	(1.34)
GDA	Tansig	Logsig	0	165	21.88	26.15	(6.70)
DS-GDA	Tansig	Logsig	6	191	18.75	19.95	(0.89)
DS-GDA	Tansig	Linear	4	70	18.75	20.36	(0.90)
QNN	Tansig	Logsig	2	116	19.27	21.41	(1.31)
MLP 6-1							
LM	Tansig	Logsig	2	58	19.27	20.99	(1.30)
GDA	Tansig	Logsig	0	202	20.31	21.67	(1.13)
QNN	Tansig	Linear	3	101	19.27	20.68	(1.04)

5 Discussion

The results as a whole show that SIANNs are able to achieve slightly better results than MLPs in these medical diagnosis problems. The performance of the trained networks, however, depends on the structure of the network as well as the training algorithm used.

Generally, the networks trained using the GDA algorithm required a longer training time and the test results achieved were not as good, especially for the more complex diabetes problem. This is to be expected, as GDA is only a first-order algorithm. In comparison, the hybrid DS-GDA achieved the best results overall for the SIANNs.

The breast cancer problem appears to be one that can be easily handled by SIANNs, with almost 100% success rate. The diabetes diagnosis

on the other hand proved to be more of a problem for both SIANNs and MLPs, with both achieving only about 80% success on average. The problems faced with the data should also be factored into this.

A point that can be taken out of these results is that one basic tenet of computer programming applies to these classifiers, and to diagnosis in general – Garbage In, Garbage Out. The neural network classifiers can only base their decisions on the given inputs, so care has to be taken to ensure that proper training data is provided. While real-life data will never be perfect, some sort of analysis of the data should be performed. If a significant portion of the input data is unavailable or incorrect, it would probably be better to eliminate these inputs. In the case of diabetes diagnosis, the removal of 2 out of the 8 inputs did not affect the performance for the classifier networks. Instead, it had the benefit of reducing the complexity of the networks used as well as the training time.

6 Conclusion

Overall, the results show that SIANNs can be applied successfully to medical diagnosis problems. SIANNs were able to consistently achieve 100% success in the breast cancer diagnosis test. For the diabetes problem SIANNs were able to achieve up to 83% success, which is better than some other published results. The success rates achieved were dependent on the structure of the network and the training algorithm used. The hybrid direct solution-adaptive gradient descent (DS-GDA) algorithm produced the best results for the SIANNs. The results depend more heavily on the actual data used for the diagnosis. The process of selecting the right inputs to be used and validity of the measurements has a greater impact on the effectiveness of these diagnosis tools than structural or algorithmic factors.

7 References

[1] A. Bouzerdoum, "A New Class of High-Order Neural Networks with Nonlinear Decision Boundaries," Proc. 6th Inter. Conf. on Neural Information Processing, Vol. III, pp. 1004–1009, Perth, 16–20 Nov. 1999.

[2] A. Bouzerdoum and R. B. Pinter, "Shunting Inhibitory Cellular Neural Networks: Derivation and Stability Analysis," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, pp. 215 - 221, 1993.

[3] R. B. Pinter, "Product term nonlinear lateral inhibition enhances visual selectivity for small objects or edges," *J. Theor. Biol.*, vol. 100, pp. 525-531, 1983.

[4] R. Beare and A. Bouzerdoum, "Biologically inspired local motion detector architecture," *J. Opt. Soc. Am. A*, vol. 16, pp. 2059-2068, 1999.

[5] A. Bouzerdoum, "Convergence of symmetric shunting competitive neural networks," in *Complex Systems: From Biology to Computation*, D. Green and T. Bossomaier, Eds. Amsterdam: IOS Press, 1992, pp. 301-312.

[6] H. N. Cheung, A. Bouzerdoum, and W. Newland, "Properties of Shunting Inhibitory Cellular Neural Networks for Colour Image Enhancement," presented at ICONIP '99, Perth, Australia, 1999.

[7] O. L. Mangasarian and W. H. Wohlberg, "Cancer diagnosis via linear programming," *SIAM News*, vol. 23, pp. 1 -18, 1990.

[8] C. L. Blake and C. J. Merz, "UCI Repository of machine learning databases," University of California, Department of Information and Computer Science, 1998.

[9] G. Arulampalam and A. Bouzerdoum, "Training Shunting Inhibitory Artificial Neural Networks as Classifiers," *Neural Network World*, vol. 10, pp. 333-350, 2000.

[10] H. Demuth and M. Beale, *Neural Network Toolbox User's Guide*, Version 3 ed: The MathWorks Inc., 1992.

[11] M. T. Hagan and M. B. Menhaj, "Training Feedforward Networks with the Marquadt Algorithm," *IEEE Transactions on Neural Networks*, vol. 5, pp. 989 - 993, 1994.

[12] F. Barmann and F. Biegler-Konig, "A learning algorithm for multilayered neural networks based on linear least squares problems," *Neural Networks*, vol. 6, pp. 127-131, 1993.

[13] A. Bouzerdoum and T. R. Pattison, "Neural Network for Quadratic Optimization with Bound Constraints," *IEEE Transactions on Neural Networks*, vol. 4, pp. 293-304, 1993.

[14] G. Arulampalam and A. Bouzerdoum, "Novel Training Algorithm Based on Quadratic Optimisation Using Neural Networks," in *Bio-Inspired Applications of Connectionism*, vol. 1, J. Mira and A. Prieto, Eds. Berlin: Springer-Verlag, 2001, pp. 410-417.

[15] L. Prechelt, "PROBEN 1 - A Set of Neural Network Benchmark Problems and Benchmarking Rules," Fakultat fur Informatik, Universitat Karlsruhe, Karlsruhe, Germany Tech. Rep. 21/94, September 1994 1994.

[16] J. Sherrah, "Automatic Feature Extraction for Pattern Recognition," Ph.D. Thesis. Adelaide: University of Adelaide, 1998.

[17] T. Waschulzik, W. Brauer, T. Castedello, and B. Henery, "Quality Assured Efficient Engineering of Feedforward Neural Networks with Supervised Learning (QUEEN) Evaluated with "Pima Indians Diabetes Database"," presented at IJCNN 2000, Como, Italy, 2000.